REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
26-09-2016	Final Report	15-Sep-2009 - 14-Sep-2013

4. TITLE AND SUBTITLE
Final Report: Computational Biomathematics: Toward Optimal Control Of Complex Biological Systems

5a. CONTRACT NUMBER
W911NF-09-1-0538

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHORS
Launbenbacher Reinhard

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
Virginia Polytechnic Institute & State Unive
North End Center, Suite 4200
300 Turner Street, NW
Blacksburg, VA 24061 -0001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
56757-MA.5

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
See attached.

15. SUBJECT TERMS
Final Report

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Reinhard Launbenbacher
UU	UU	UU	UU		19b. TELEPHONE NUMBER 540-231-2100

Standard Form 298 (Rev 8/98)
Prescribed by ANSI Std. Z39.18

Report Title

Final Report: Computational Biomathematics: Toward Optimal Control Of Complex Biological Systems

ABSTRACT

See attached.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received		Paper
08/25/2011	1.00	Franziska Hinkelmann, Madison Brandon, Bonny Guang, Rustin McNeill, Grigoriy Blekherman, Alan Veliz-Cuba, Reinhard Laubenbacher. ADAM: Analysis of Discrete Models of Biological Systems Using Computer Algebra, BMC Bioinformatics, (01 2011): 0. doi: 10.1186/1471-2105-12-295
08/25/2011	2.00	David Murrugarra, Franziska Hinkelmann, Abdul Salam Jarrah, Reinhard Laubenbacher. A Mathematical Framework for Agent Based Models of Complex Biological Networks, Bulletin of Mathematical Biology, (9 2010): 0. doi: 10.1007/s11538-010-9582-8
08/25/2011	3.00	D. Murrugarra, R. Laubenbacher. Regulatory patterns in molecular interaction networks, Journal of Theoretical Biology, (12 2011): 0. doi:
TOTAL:		3

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received	Paper
TOTAL:	

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received Paper

TOTAL:

Number of Manuscripts:

Books

Received	Book
09/28/2012 4.00	Franziska Hinkelmann, Matt Oremland, Reinhard Laubenbacher. Agent-based models and optimal control in biology: a discrete approach. , Amsterdam: Elsevier, (2012)

TOTAL: **1**

Received	Book Chapter

TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

NAME PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Under Graduate students supported

NAME PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: ......

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:......

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:......

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):......

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:......

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense ......

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields: ......

Names of Personnel receiving masters degrees

NAME

Total Number:

Names of personnel receiving PHDs

NAME

Total Number:

Names of other research staff

NAME PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

Technology Transfer

Model reduction

In order to get reliable results from agent-based models, simulations typically have to be run many times and results averaged, given that while the rules determining agent behavior in a model may be fixed, they often involve random processes (e.g., movement in a random direction). The first step towards model reduction involves figuring out how many simulations must be run (for fixed parameter settings) in order to obtain reliable results; this value may be model dependent. When we refer to data obtained from the model, we mean data obtained from this pre-determined number of repeated simulations.

The first step towards model reduction is to reduce the physical landscape of the model, scaling the number of each agent type accordingly. We collect data at each physical size, and compare results. In some models there exist what are known as 'phase transitions', where certain phenomena occur which may not occur at different sizes. The goal of this phase of model reduction is to determine how much we can reduce the physical landscape without introducing or removing any relevant phase transitions, because the reduced landscape will result in faster run times. It is possible to think of such spatial reductions as aggregations rather than a smaller 'world'. In other words, if we reduce from 100 agents to 25, we can either think of this as a smaller sample size, or we can interpret it as if we are putting our agents into groups of 4 and evaluating them as groups instead of individuals.

We use Pearson's correlation coefficient on the data to determine exactly how small we can scale our model and still retain qualitatively similar behavior. We correlate the data at a reduced size with the data from the original size of the model, and set a benchmark – for example, we may require that the reduced data correlate at a value of 0.95 or higher in order to be said to be within the bounds of acceptability. In this way, we use a statistical measure to determine the accuracy of our reduction instead of relying on guesswork or intuition.

This reduction technique was applied both to the Rabbits and Grass model (which is spatially homogeneous) and to the SugarScape model (which is spatially heterogeneous). In both cases we were able obtain results in approximately 25% of the original run time, without the loss or addition of phase transitions. Code was written for NetLogo that automatically reduces a spatially heterogeneous landscape to the user's desired size, allowing the user to choose the reduction algorithm to be used (nearest neighbor or bi-linear interpolation).

The following paper is in preparation: Scaling methods and heuristic algorithms for agent-based models. Matt Oremland, Reinhard Laubenbacher.

SugarScape

The SugarScape model was subjected to the model reduction techniques described above as a test subject of a spatially heterogeneous model. However, a different technique was developed as well – that of converting the model dynamics to a series of difference equations, which could then be subjected to control and thereby replace simulation.

The model consists of four categories of agents, and four distinct spatial regions. Each agent has an energy level from zero to fifty. Thus for each category, 200 difference equations were implemented, one for each energy level in each spatial region. For example, the equation S2,13(t) keeps track of how many agents there are in region 2 with energy level 13.

The bulk of the work for this approach is to determine exactly what these equations should be. In general, they are dependent both on the rules of the agent-based model and on the information one needs to obtain. In this case, we were interested in taxing agents based on their energy level and spatial location, so we needed to keep track of both variables. If one were interested only in spatial location, the number of equations could be substantially lowered.

Since the equations depend on what information we are interested in, automatic conversion of agent-based models to systems of equations seems daunting. However, we are currently working on parameter estimation methods that show some promise. In this approach, we generate data from the simulation and attempt to determine parameters that fit the equations to the data. This work has been successful with the Rabbits and Grass model and is currently underway for a more spatially complicated version of SugarScape.

Paper in preparation: Mathematical conversion and analysis of agent-based models. Matt Oremland, Reinhard Laubenbacher.

Control

Control for ABMs has taken one of two forms: when not converting the model, we apply our model reduction techniques and use data from simulations in order to determine the cost of a particular control. This can be computationally expensive; on the other hand, this strategy can be applied to virtually any model, no matter how complicated. The only concession one must make is that the cost is determined as an average value from a repeated number of simulations, and is not able to accommodate single runs while retaining accuracy.

The actual method of control used is in the form of heuristic algorithms. In general, these algorithms search through a virtually infinite set of controls and evaluate them, then incorporate some method of refining the controls in order to obtain even better results. Evolutionary algorithms combine two control choices, for example, hoping to get the best features of each. Others, such as simulated annealing, refine a single control rather than working with a population of controls.

The drawback of these methods are that they cannot be guaranteed to determine a globally optimum control, and that there is no way to

determine ahead of time which algorithm will work best for a given model. On the other hand, the results are far better than those obtained by random search (as we have shown in the course of this work). In addition, given the typical interest in agent-based models as simulations of real-world systems, one may be satisfied with controllers that are better than any previously known, rather than requiring that they be provably optimal.

We have implemented the following heuristic algorithms: random-mutation hill climbing, simulated annealing, genetic algorithms, and Pareto optimization. In general, genetic algorithms returned the best results in most cases. Pareto optimization is a means of multi-objective optimization, wherein one does not have to determine a cost function ahead of time, but rather only specify the variables of interest. For example, if we wish to determine a controller that reduces cost and maximizes efficiency, Pareto optimization allows us to conduct a search without specifying which of those two aims is the priority, or having to assign specific weights to them. Rather, this method returns a suite of controls, each optimal under different conditions. This allows the user to specify priorities, and change them without having to re-run the algorithm.

For models wherein difference equations have been obtained, certain information is more easily obtained: we can solve numerically for steady states, and perform stability analysis. However, the control methods we have examined are currently limited to heuristic algorithms. The main advantages of model conversion are the drastic reduction in run time, and the precision of the model description (which is an ongoing issue in the agent-based modeling community). We are also working on methods which will take advantage of mathematical theory in order to not have to perform heuristic searches on the difference equations, but rather exploit known techniques for optimal control of discrete equations.